一种基于轻量级深度网络的目标识别方法 *

李亚辉，刘　俊

(杭州电子科技大学 通信与传输信息融合国防重点学科实验室，杭州 310018)

摘　要：针对资源受限条件下目标识别任务，提出了一种基于轻量化深度网络的目标识别方法。通过优化卷积操作、模型参数压缩、增强特征表达深度等网络结构的优化方法，设计并实现了针对嵌入式平台应用的轻量化网络模型结构，使得深度网络模型在保证精度的条件下，实现了模型参数和运行所需资源的大幅缩减。实验表明，提出的轻量化深度模型较 ILSVRC-15 冠军提出的基础模型 ResNet，能够实现在 ImageNet-67 数据集上，网络模型压缩为基础模型 10.2%的条件下仍保持 93.5%的目标识别准确率。

关键词：深度学习；目标识别；轻量化；嵌入式应用

中图分类号：TP183　　doi: 10.19734/j.issn.1001-3695.2018.07.0663

Object recognition method based on lightweight depth network

Li Yahui, Liu Jun

(*Fundamental Science on Communication Information Transmission & Fusion Technology Laboratory, Hangzhou Dianzi University, Hangzhou* 310018, *China*)

Abstract: Aiming at the task of object recognition under resource constrained condition, this paper proposed a method of object recognition based on light weight depth network. By optimizing the design method of the network structure such as convolution operation, model parameter compression and enhancement of feature expression depth, this paper designed and implemented the lightweight network model structure named Se-DResNet for embedded platform. So that the depth network model can reduce the parameters of the model and the resources needed for operation under the condition of guaranteeing the precision. The experimental results show that the lightweight depth model has better performance than that of the basic model proposed by ILSVRC-15 champion ResNet, and it can achieve the model accuracy of 93.5% under the condition that the model with 10x fewer parameters on IMAGENET-67 data set.

Key words: deep learning; object detection; lightweight; embedded application

0　引言

军事领域中自动目标识别（automatic target recognition，ATR）作为战场环境感知领域的重要环节，是一种利用各种传感器，从客观世界获取目标/背景信号，并使用计算机信息处理方法自动地分析场景信号、检测、识别感兴趣的目标及获取目标各种定性、定量性质的军事目标识别方法。作为光谱信息领域关键技术的图像目标类别检测技术，其通过从原始的图像中提取目标的特征并以此为基础完成目标的识别任务，也是计算机视觉中低层视觉处理的关键问题。图像目标类别检测技术又称类别级目标检测或目标检测，旨在利用图像处理与模式识别等领域的理论和方法，检测出图像中存在的目标对象，确定这些目标对象的语义类别，并使用边界框标定出目标对象在图像中的位置[1]。图像目标类别检测是目标分类的一个子问题，目标分类可以分为三个层级：图像级，即确定图像中是否有相关的目标对象，如图像分类、图像注释技术；区域级，即确定图像中某个区域是否含有某个类别，即本文所述的图像目标类别检测；像素级，即确定图像中各个像素归属于哪类目标对象。像素级也可以细分为类别级目标分割和语义分割两类。在此，本文只关注在第一层级，也即是图像级目标识别任务的研究。

随着物联网、大数据的飞速发展，智能终端设备（intelligent of terminal，IoT）在人们的生活中发挥着越来越重要的作用，同时也为智能感知、万物互联的智能化时代迎来了新的契机，也为硬件资源、数据量强依赖的深度学习方法提供了基础支撑。深度学习，由于其强大的特征提取和表达能力，使得深度学习在目标检测、语音识别、自然语言处理等领域中发挥着越来越重要的作用。传统的计算机视觉任务通常采用特定领域专家设计的针对特定目标类别的特征，其过程具有特征设计的复杂性和低效性，难以实现在其他领域中得以较好地使用。而深度学习采用在权值迭代更新过程中实现对图像空间域信息的提取和表达，从而能够实现自动提取出目标特征，省去了传统目标识别算法所使用的特征需要领域专家针对特定类别进行抽象特征设计的繁杂工作。特别是迁移学习的出现使得计算机视觉任务的设计者可以不过度关注特征结构设计这一繁杂过程，这也使得深度网络可以实现在不同领域之间快速地迁移，而不需要重新设计网络模型结构。

然而深度学习算法的使用也对硬件的计算能力和样本数据的数量和质量提出了更高的要求。深度网络模型，由于其内部庞大的参数量和多级的非线性映射，使得模型具有良好的特征提取和表达能力。此外，由于特征提取过程采用基于样本集的迭代学习过程来完成，所以该过程需要满足硬件计算能力和样本集的体量两大要求。而这两者也正是深度学习

收稿日期：2018-07-19；**修回日期：**2018-09-14　**基金项目：**国家自然科学基金重点资助项目（61333009）；海军装备预研创新项目

作者简介：李亚辉（1992-），男，河南周口人，硕士研究生，主要研究方向为信息融合、目标识别、网络压缩（yahuiliem@163.com）；刘俊（1971-），男，教授，博士，主要研究方向为信息融合、模式识别、智能系统.

在大数据时代能够再次复苏的两个重要因素。

经典的网络模型中由于存在大量的可学习参数，使得模型需要大量的样本数据，从而保证网络模型具有较好的泛化能力。但由于其密集的计算和存储使得模型的应用范围受到了严重的制约，使得其较多地应用在服务器等高性能集群上。然而令人欣喜的是世界各地的研究者在近几年的研究中发现，深度网络虽然需要较多的神经元以实现复杂的非线性映射的拟合能力，但其由于结构设计等因素的存在，使得网络模型中中存在大量的冗余参数，而删除这些参数并不会对原有网络模型的性能产生较大的影响。因此本文针对军事领域中无人机等硬件资源受限条件下深度网络的轻量化网络模型设计进行了研究，基于经典的残差网络模型结构 ResNet 进行了针对卷积优化、参数压缩和增强特征表达等方面的优化操作，实现了在 ImageNet-67 数据集上，模型压缩为原有 10.2%的条件下仍保持 93.5%的目标识别精度。

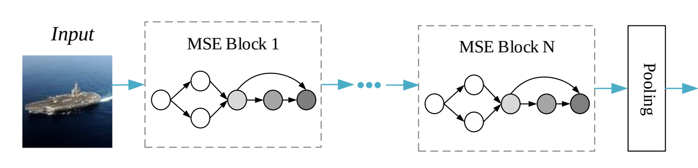


图 1　轻量化目标识别网络整体结构

Fig. 1　Structure of lightweight object recognition network

1　相关工作

传统的计算机视觉任务采用经典的数字图像处理的方式进行，通过设计相关特征提取算子从而实现图像中特定目标特征的提取，并通过支持向量机（support vector machine，SVM）等分类器完成目标识别任务。但通过特征提取算子的设计需要特定领域的专家知识作为支撑。同时，针对特定领域的算子设计并不能方便地在其他领域进行应用，这也使得传统的计算机视觉任务发展较为缓慢。

2012 年以 AlexNet 为首的五层深度网络为计算机视觉领域开辟了新的道路，同时初步展现出了深度网络强大的性能[2]。在此之后 ZF、VGG、GoogleNet、ResNet、DenseNet，以及各种优化网络模型被提出，网络模型的性能也在逐步提升。各模型性能对比如图 2 所示[3~16]。然而，随着模型性能的提升，深度模型训练所需的硬件资源从最初的普通计算机到如今的大规模高性能服务集群，样本数据集也从 CIFAIR-10、CIFAIR-100 到如今的 ImageNet-1k 以及 Open Images，图片数据量也由最初的 60 000 幅到如今的 1 900 000 幅之多。如此大规模的密集计算使得深度模型难以实现在硬件资源受限条件下进行有效的部署，因而有越来越多的研究者开始转向深度网络模型压缩这一新的领域，旨在保证模型性能的条件下使得模型的参数量和计算量实现大规模缩减，从而减少模型训练和部署所需的样本数据量和硬件资源。

2016 年，MIT 博士生韩松通过对深度网络结构进行深入研究之后提出了一种基于裁剪、量化和霍夫曼编码的深度网络压缩方法，实现了模型性能与 AlexNet 相同的条件下模型大小压缩为原有的 1/510。并由此引起了世界各地的研究学者开始针对网络模型结构优化的网络压缩方法的研究，其中较为著名的有 MobileNet、ShuffleNet 以及 Google 针对 MobileNet 基础上进行优化后提出的 MobileNet v2 等针对嵌入式等硬件资源受限条件下的轻量化网络结构[8]。这些网络模型从深度网络结构的设计角度进行研究，通过优化深度网络模型中神经元之间的连接方式，在保证模型性能的前提下实现了模型参数和计算量的大幅度缩减。参数量的缩减随之带来的就是硬件要求的降低，这也使得深度网络模型能够逐渐在实际的工业环境中得以实现大规模的应用。

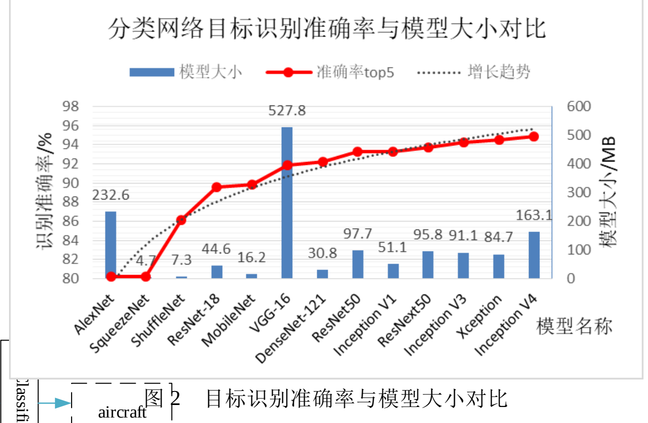


图 2　目标识别准确率与模型大小对比

Fig. 2　Comparison of accuracy and model size of object recognition

在实际的工业应用环境中尤其是 IoT 设备，硬件资源的计算和存储能力都较大规模计算机群存在巨大的差距。表 1 统计了目前较为常见的硬件设备的计算能力数据。

表 1　常见显卡指标数据

Table 1　Common graphics card data

显卡名称	显存/G	处理能力/Tflops
GeForce GTX 1080	8	8.2
GeForce GTX 1080 Ti	11	10.6
Nvidia TITAN X	12	10.2
Nvidia TITAN Xp	12	10.8
GeForce GTX TITAN	12	4.5
K80 GOU Accelerator	12	5.6~8.8
Jetson Tegra K1	**2**	**0.326**

从表中可以看出，常规的 IoT 设备通常的计算能力仅为大规模服务集群使用的 TITAN Xp 计算硬件的 3%，并且存储能力仅为 16.7%，如此严峻资源受限条件也对于复杂深度网络的应用提出了更高的要求，也使得针对深度网络模型优化的研究更为迫切。因此，如何更好地实现针对嵌入式等硬件资源受限条件下深度网络的应用的问题的研究便显得尤为重要。

2　算法总体框架

深度网络通常采用减少模型的参数量和单个参数存储空间以此优化网络模型的深度网络压缩来解决。然而简单地通过删除网络模型的结构参数而不进行结构的调整优化，将会导致模型的性能产生大幅度下降，从而影响网络模型的实际使用。本文关注的目标识别方法为第一个层级，也即图像级目标识别问题。通过借鉴历年 ILSVRC 竞赛中冠军网络模型的思想，并结合嵌入式应用的需求设计了基于模块化的 Se-DResNet 网络模型模型总体结构，如图 1 所示。

本文提出的深度网络模型结构在设计中采用模块化设计方式，通过将各个模块按照拓扑结构进行有序的连接，最终形成模型的整体结构。Se-DResNet 模块结构如图 3 所示。其中左图为模块输入和输出特征图维度相同，而右图为将输入特征图维度进行缩减时使用。

从模块结构示意图中可以看出，在上一级模块或者数据读取层输出到本层模块时，首先将原有特征通道分为两个相同的分支。其中一个分支使用 shortcut connection 思想，将其直接作为模块输出的前有一层 $f_i(x)$，而另一个分支则采用 bottleneck 思想首先将特征通道进行压缩，其中压缩比例 w_f

可以根据实际情况进行设定。

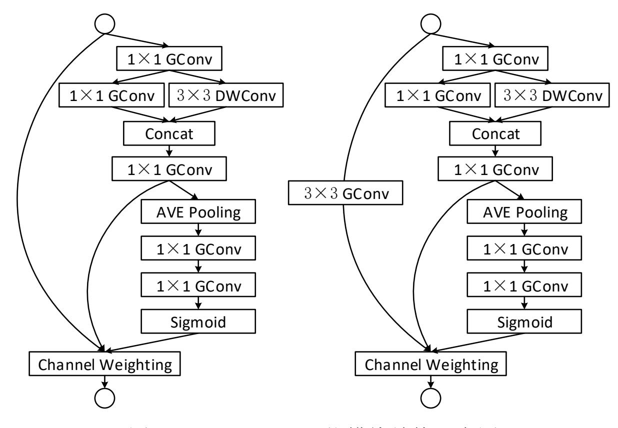


图 3　Se-DResNet 网络模块结构示意图

Fig. 3　Diagram of module structure of Se-DResNet

在 bottleneck 输出后将总的特征通道的数量按照 w_s 比例系数进行通道切分，其分别通过不同尺度的卷积核进行卷积特征提取操作，在此使用 1×1, 3×3 尺寸的卷积核以此来实现不同特征通道之间感受野的不同，从而提升小网络的特征表达能力。并在输出不同卷积核输出的特征图进行拼接操作，使其变为一个完整的特征通道，在此之后通道 1×1 的卷积操作实现不同感受野特征通道之间的特征融合。

由于深度网络存在较大的冗余特征，在 1×1 卷积输出后不同特征通道的特征质量也不尽相同，所以在此采用基于特征通道加权思想的卷积操作流程，其通过将单个卷积通道变换为一个实数数值，以此来衡量特征通道的质量优劣，并将该实数值作为通道权值进行各通道特征之间的加权操作，输出特征图 $f_{se}(x)$ 。最终将 $f_i(x)$ 和 $f_{se}(x)$ 通过残差网络思想将其进行特征逐点相加最终输出该模块处理后的特征信息到下一特征提取模块或者模型的分类器实现样本目标类别的识别。

该模型结构基于 ResNet 网络本文主要针对优化卷积操作、模型参数压缩和增强特征表达三个方面进行改进，下面将对每个改进方法进行详细的阐述。

2.1　优化卷积操作

标准的卷积操作是将输入层的特征通道与卷积核进行计算处理后加和输出结果，作为下一层的特征输入。标准卷积公式如式（1）所示。

$$g(x,y)=\sum_{s=-a}^{a}\sum_{t=-b}^{b}w(s,t)f(x+s,y+t) \tag{1}$$

其中： w 为卷积核； f 为特征图。

使用标准的卷积计算虽然能够完成图像特征的提取，但是由于标准卷积核的计算需要根据输入通道和输出通道来决定卷积核的数量，所以三者之间存在着较强的耦合性。如果仅仅从输入或者输出通道角度进行优化，由于其耦合关系的存在难以实现较好的效果，所以在此使用深度可分离卷积对卷积操作进行优化，其通过将标准的卷积操作划分为 Depthwise 和 Pointwise 两个部分进行。其流程示意图如图 4 所示。

在标准的卷积操作中，假设输入的特征图的尺寸为 $M\times D_k\times D_k$ ，其中： M 为输入特征图的数量； D_k 为输入特征图的尺寸。输出特征图的尺寸为 $N\times D_f\times D_f$ ，其中： N 为输出特征图的数量； D_f 为输出特征图的尺寸，则标准卷积操作所需的计算量为计算公式如式（2）所示。

$$C_{fs}=D_k\times D_k\times M\times N\times D_f\times D_f \tag{2}$$

而通过深度可分离卷积操作之后，输入和输出特征图的尺寸均相同的情况下，其计算量可以分为深度卷积计算量和逐点卷积计算量两部分。其中所需计算量为计算公式如式(3）所示。

$$C_{fd}=D_k\times D_k\times M\times D_f\times D_f+M\times N\times D_f\times D_f \tag{3}$$

图 4　深度可分离卷积示意图

Fig. 4　Diagram of depth separable convolution

通过将标准卷积和深度可分离卷积进行比较可以发现，深度可分离卷积在模型计算量的减少方面，较原始标准卷积减少比例计算公式如式（4）所示。

$$\gamma=\frac{C_{fd}}{C_{fs}}=\frac{D_k\times D_k\times M\times D_f\times D_f+M\times N\times D_f\times D_f}{D_k\times D_k\times M\times N\times D_f\times D_f}=\frac{1}{N}+\frac{1}{D_k^2} \tag{4}$$

在深度网络中 1×1 卷积承载着多特征通道之间的信息融合，但是由于点卷积操作同样是采用遍历的方式进行，虽然模型的参数量较少，实际的计算量却仍然较为庞大，所以本文借鉴分组卷积的思想，将网络中 1×1 的卷积操作全部使用组卷积实现。但是本文并没有在组卷积之后使用通道混排操作，其原因有以下两点：

a) 由于在前三次组卷积操作的过程中使用的通道数不相同，在通道数量不同的情况下分组卷积操作本身就自带有通道重排的操作，所以不会出现边缘效应。

b) 使用通道混排操作将会增加网络的计算量，同时增加网络在运行过程中所使用的内存空间，这对于原本硬件资源受限的条件的设计要求相违背，也会导致模型的实时性能下降。

2.2　模型参数压缩

为了增加网络中各层感受野的丰富程度，同时有效降低模型的参数量，在此使用基于 Bottleneck 思想的模型压缩方法。首先通过组卷积的方式将输入 1×1 点卷积和 3×3 可分离卷积的特征通道数量进行压缩，压缩比例使用 w 来控制；然后根据实际的应用需要设定输入 1×1 和 3×3 可分离卷积的特征通道的数量，两者的比例使用 r 来进行控制。在实际应用中可以根据精度和实际硬件的需求对 w,r 两个参数进行选择。其示意图如图 5 所示。

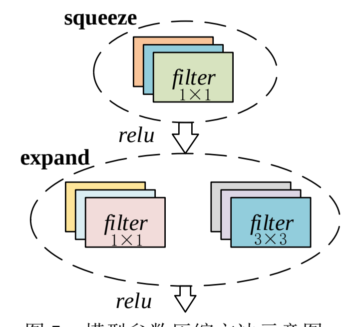


图 5　模型参数压缩方法示意图

Fig. 5　Diagram of model parameter compression method

2.3　增强特征表达

感受野的丰富能够提升模型的性能，究其原因是由于感受野的丰富为模型提取特征的质量提供了保障，然而如果将优质的特征进行充分的利用能否进一步提升网络的性能是另一个方法。针对嵌入式应用设计的网络其特点是模型的参数

量和模型运行所需的硬件要求都比较低，但是这种网络结构也往往有一个问题存在，那就是模型的特征表达能力不足，这也导致模型的性能并不如服务器集群上部署的网络结构性能的主要原因。因此本文针对轻量化网络特征表达能力较弱的问题，采用 SeNet 中提出的基于通道加权的思想来提升网络模型对特征的表达能力，其原理如图 6 所示。

从示意图中可以看出，该方法并不是基于空间维度来进行模型结构的优化，如 Inception 等，而是从特征通道之间的关系角度来进行模型结构的优化，通过显式地建模通道之间的相互依赖关系，自适应地重新校准通道的特征响应。

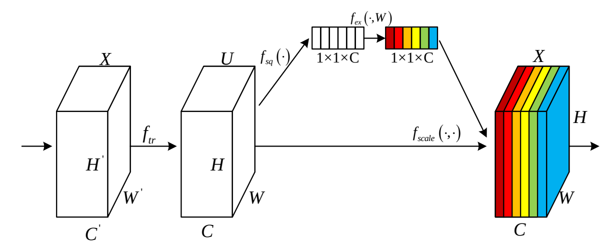


图 6　通道加权示意图

Fig. 6　Diagram of channel weighting

具体来说，其采用将原始特征通道首先经过全局池化操作转变为 1×1×C，然后使用非线性激活函数得出每个通道权重的比例值，最后通过将该比例值映射到原有特征通道的每一个特征值上得到最终的特征输出。输出的特征图由于是基于每个特征图对应的系数加权之后的结果，所以在一定程度上起到了自动特征差异化处理的过程，有利于目标关键特征的提取。

2.4　重塑损失函数

由于数据集制作以及数据预处理阶段，各类别样本数量以及单类别中难分易分样本的不同等问题，使得输入分类器的样本可能存在样本比例失衡的问题。

当某一类别在数据集中存在较多时分类器往往能够针对该类别获得比较好的性能，但传统的交叉熵损失函数在模型训练的过程中针对已经能够较好识别的类别并没有进行区分，使得分类器在训练的过程中重复性地对已经获得较好的识别性能的样本仍然进行着学习，这也使得分类器难以专注于难以识别样本的学习，这将增加模型的训练时间。因此本文采用基于难识别样本挖掘思想的 RetinaNet 中提出的焦点损失函数作为识别网络训练的损失函数。焦点损失函根据网络预测输出的每个样本的类别的概率值作为损失函数的权重。焦点损失函数表达式如式（5）所示。

$$FL(p_t) = -\alpha_t (1-p_t)^{\gamma} \log(p_t) \tag{5}$$

其中：$(1-p_t)^{\gamma}$ 为调制系数，用于控制不同样本对损失函数的贡献率；当 γ=0 时便是标准的交叉熵损失函数；p_t 为模型在训练过程中预测输出的样本类别的概率。

3　实验结果与分析

3.1　实验环境简介

本文实验环境分为网络训练和网络测试两个部分。其中网络训练部分是基于 TITAN X 的硬件资源平台，使用 Xeon E5-2450@2.00 GHz CPU 主板,板载内存 16 GB；软件环境为 Ubuntu 14.04 系统，Caffe 深度学习框架。网络测试部分验环境为基于英伟达 Jetson TK1 为硬件基础平台，其板载内存（显存）2 GB；软件环境为 Jetson TK1 定制版的 Ubuntu 14.04，Caffe 深度学习框架。

3.2　网络模型参数设置

在上述提出的轻量化深度网络模块的基础上，在实际使用中可以根据硬件环境的不同，通过设置 w, r 的值进行模型参数量的调整，在此使用 $w_f = 1.0, w_s = 0.5$。通过将各个模块进行连接，最终形成实验所需的网络模型结构。模型结果参数设置详细信息如表 2 所示。其中 MB 表示 Match_Block，DB 表示 DRes_Block。

整个网络模型采用一层普通的卷积层之后直接使用了池化操作，用于实现原始图像的模糊化突出图像中目标的轮廓信息；之后使用四组“DB+MB”的结构完成在前一层特征图的基础上提取更为高层的语义特征信息；最终使用 softmax 分类器完成特征的分类，识别出目标的类别信息。

表 2　轻量化模型结构参数设置（ImageNet）

Table 2　Lightweight model structure parameter setting (ImageNet)

层名称 (*layer*)	输出维度 (*output size*)	卷积核 (*kernel size*)	步长 (*stride*)	重复次数 (*repeat*)
Image	224×224	–	–	–
Conv1	112×112	7×7	2	1
Pool1	57×57	3×3	2	1
MB1	57×57	3×3	1	1
DB1	57×57	3×3	1	1
MB2	29×29	3×3	2	1
DB2	29×29	3×3	1	2
MB3	15×15	3×3	2	1
DB3	15×15	3×3	1	4
MB4	8×8	3×3	2	1
DB4	8×8	3×3	1	2
Pool2	1×1	–	*global*	1

在完成模型构建的基础之上，需要使用合理的深度网络优化算法以及恰当的网络模型训练参数对上述的网络模型结构进行优化，从而实现模型中神经元之间连接参数的更新优化。模型训练的参数设置如表 3 所示。在此使用 SGD 优化算法作为模型优化的计算方式。

表 3　模型训练的参数设置

Table 3　Model training parameter setting

名称	参数
基础学习率（base lr）	0.01
优化策略（type）	SGD
学习率改变策略（lr policy）	multistep
最大迭代次数（max iter）	450 000
学习率变化比例（gamma）	0.1

3.3　实验结果与分析

为了较快验证算法的可行性以及考虑到实际军事目标识别应用的针对性，本文针对 ImageNet-1k 数据集进行了类别删减，将与军事目标无关的类别进行删除，最终只保留与军事目标相关的 Person、Boat 等在内的 67 个目标类别数据。为了保证对比的公平性，以下实验结果数据均是基于该数据集进行。

本文主要完针对以下实验进行了对比工作：

a) 原始 ResNet 模型与本文提出的 Se-DResNet 模型对比。

b) Se-DResNet 与现有轻量化网络模型 MobileNet v1、MobileNet v2、ShuffleNet、ResNet-18 网络结构进行了对比。

首先将本文提出的 Se-DResNet 网络模型与 ResNet 进行了对比，通过该实验数据来检验本文提出的深度网络结构与基础网络模型结构的性能的差别。对比实验数据如表 4 所示。其中@.5 表示通道裁剪为原有的 0.5 倍，其他依此类推。

表 4　ImageNet-67 数据集上模型尺寸 vs 准确率

Table 4　Model size vs accuracy on ImageNet-67 data set

模型名称	大小/MB	压缩比/%	准确率/%	误差/%
ResNet-50	97.7	–	94.4	–
Se-DResNet	**10.1**	**10.3%**	**93.5**	**-0.9**
ResNet-50@.5	64.3	65.8%	93.8	-0.6
Se-DResNet@.5	7.3	7.4%	92.3	-1.9
ResNet-50@.25	16.3	16.7%	92.9	-1.5
Se-DResNet@.25	**1.7**	**1.7%**	**89.9**	**-4.5**

从表中可以看出，在不进行任何模型裁剪的条件下，原始的 Se-DResNet 网络模型大小仅为 10.1 MB，约为 ResNet-50 网络模型大小的 1/10。但模型的性能与 ResNet 模型仅相差 0.9%，在进一步压缩的条件下，网络模型能够实现在压缩为原有结构的 7.4%的条件下，也即是比 Se-DResNet@.5 更小的模型大小的条件下，仍能实现比 Se-DResNet@.5 高 0.5%的目标识别准确率。而整个模型在运行阶段所需的显存占用也存在巨大的差别，Se-DResNet 网络模型仅需要使用 533 MB，较原始 ResNet 网络显存占用缩减了 85%。

考虑到实际嵌入式平台应用的需求，本文针对原始 Se-DResNet 网络进行了基于通道的深度裁剪，以此检验模型是否能够在不同硬件资源受限条件下的应用需求。从表 5 中可以看出，但当模型裁剪为原有的 1.7%时模型性能依然能够得到有效地保证，并且能够实现在与 ShuffleNet 模型相近的目标识别准确率，但整个模型的存储空间占用仅为 1.7 MB。

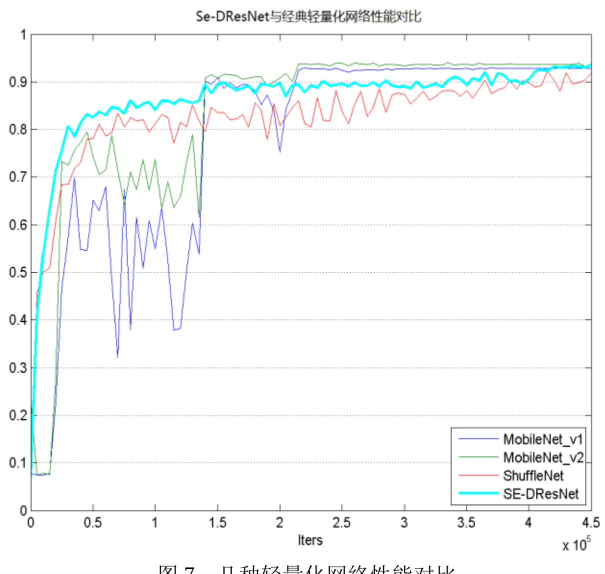


图 7　几种轻量化网络性能对比

Fig. 7　Performance comparison of several lightweight networks

在完成与 ResNet 对比的基础上，本文又针对嵌入式应用的需求，与 ResNet-18、MobileNet v1、MobileNet v2、ShuffleNet 这一针对嵌入式应用设计的网络模型进行了对比，模型性能对比曲线如图 7 所示。

从曲线可以看出，Se-DResNet 网络结构在模型训练的初期具有较高的收敛速度，并且性能优于其他现有的针对嵌入式应用设计的网络结构。在模型训练的末端，四种轻量化模型性能收敛到近似相等，但仍有一定的差别，ShuffleNet 网络在模型训练的末期仍具有较大的性能波动，并且其性能要略低于 MobileNet v2 和 MobileNet v1 网络，而本文提出的 Se-DResNet 网络结构在最终的网络性能上实现了略高于 MobileNet v1，近似 MobileNet v2 的准确率，相差仅为 0.1%，但是模型的大小和收敛速度方面却明显优于其他几种网络模型。详细的模型性能数据信息如表 5 所示。

表 5　各种轻量化网络模型性能对比

Table 5　Performance comparison of various lightweight network models

模型名称	模型大小/MB	准确率 top1/%	准确率 top5/%
ResNet-18	44.6MB	70.8	92.1
ShuffleNet@1g3	7.3MB	69.9	90.1
MobileNet	16.2MB	70.8	92.8
MobileNet v2	14.2MB	71.6	93.6
Se-DResNet	10.1MB	71.4	93.5

在完成模型整体性能对比的基础上，本文又针对模型的每种改进进行了切割实验，以此来验证模型设计过程中每种改进对于最终模型性能的影响。切割实验数据如表 6 所示。

从表中可以看出，算法模型仅适用深度可分离卷积对原始卷积操作进行优化的情况下，由于深度可分离卷积采用将标准卷积拆分为两部操作的方式，虽然在一定程度上减少了模型的参数和计算量，但是随之带来的是模型中各个特征通道之间的信息交叉关联次数变少，所以导致了模型性能较原有 ResNet 模型性能大幅度下降。

表 6　切割实验对分类网络模型的影响

Table 6　Influence of cutting test on performance of classification network model

	Se-DResNet(1.0×)					
Depthwise convolution?	√	√	√	√	√	
Squeeze expand?		√	√	√	√	
Squeeze excitation?				√	√	
Focal loss?					√	
Accuracy	86.5	87.3	88.4	89.3	91.8	93.5

通过加入 Squeeze-expand 结构，利用不同的卷积核尺寸所产生的不同大小的感受野来增加网络模型中感受野尺寸的丰富程度，从而使模型对于不同尺寸的目标能够有效识别，从而提升网络的性能。实验结果显示，通过该方法的加入，模型性能得到了 0.8%的提升。

感受野的丰富能够提升模型的性能，究其原因是由于感受野的丰富为模型提取特征的质量提供了保障，然而如果将优质的特征进行充分的利用能否进一步提升网络的性能是另一个方法。因此，在此本文借鉴 SE 网络中所提出的基于通道加权思想的 Squeeze Excitation 网络结构，通过该结构提升小模型的特征表达能力。实验结果表明，通过该结构能够大幅提升小模型的性能，使得小模型仍然具有较强的特征提取和表达能力。

在损失函数方面，由于输入分类器的样本的质量将直接影响分类器的训练的最终性能，在大量的数量进行网络的训练过程中存在较多的冗余计算的情况，例如，数据集中的某张图片在训练过程中已经能够将其以较高的置信度将其识别出来，然而由于网络不具有筛选数据集的功能，所以导致模型重复较多的简单样本的训练，导致模型不能够针对难分样本进行专注学习。因此在此本文借用 RetinaNet 网络中所提出的焦点损失函数作为分类器的损失函数，通过算法预测的概率值动态调整损失值得权重，从而能够保证网络更加专注于难分样本的学习，也使得模型的最终性能得以提升。实验结果表明，通过焦点损失函数（focal loss）,模型性能提升了 1.7%。

4　结束语

本文针对嵌入式等资源受限条件下轻量化网络模型结构设计进行了深入研究，并在此基础上提出了 Se-DResNet 轻量

化网络结构，针对设计的轻量化网络结构进行了实际的实验验证和分析。实验证明，该轻量化网络结构实现了在保证模型准确率的基础上大幅度缩减模型参数，从而实现模型密集型计算的计算量的缩减以及部署时内存和硬盘存储空间的占用，使得嵌入式等硬件和能耗资源受限条件下模型能够实现有效部署。但本文提出的轻量化网络结构现阶段仅仅是针对图像分类任务。同时本文所使用的网络压缩方法仅仅关注在网络模型结构的优化方面，对于权值量化共享以及编码等方式并未涉及，在今后的研究中希望通过将该轻量化网络结构与目标检测框架进行有效整合，并利用除结构优化之外的深度网络压缩方法，从而实现在嵌入式等硬件平台的目标检测任务，为军事领域等提供有效的侦查手段，扩展战场领域的感知范围，为指挥决策提供基础支撑。

参考文献：

[1] 蔡强，刘亚奇，曹健，等. 图像目标类别检测综述 [J]. 计算机科学与探索, 2015, 9 (3): 257-265. (Cai Qiang, Liu Yaqi, Cao Jian, *et al.* Overview of image object category detection [J]. Journal of Frontiers of Computer Science & Technology, 2015, 9 (3): 257-265.)

[2] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks [C]// Proc of International Conference on Neural Information Processing Systems. [S.l.]:Curran Associates Inc. 2012: 1097-1105.

[3] Zeiler M D, Fergus R. Visualizing and understanding convolutional networks [C]// Proc of European conference on computer vision. Berlin: Springer, 2014: 818-833.

[4] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition [EB/OL]. (2015) [2018-07-20]. https://arxiv. org/abs/1409. 1556. pdf.

[5] Szegedy C, Liu Wei, Jia Yangqing, *et al.* Going deeper with convolutions [C]// Proc of IEEE conference on Computer Vision and Pattern Recognition . Piscataway, NJ: IEEE Press, 2015: 1-9.

[6] He Kaiming, Zhang Xiangyu, Ren Shaoqing, *et al.* Deep residual learning for image recognition [C]// Proc of IEEE conference on Computer Vision and Pattern Recognition . Piscataway, NJ: IEEE Press, 2016: 770-778.

[7] Huang Gao, Liu Zhuang, Weinberger K Q, *et al.* Densely connected convolutional networks [C]// Proc of IEEE conference on Computer Vision and Pattern Recognition . Piscataway, NJ: IEEE Press, 2017: 3.

[8] Iandola F N, Han Song, Moskewicz M W, *et al.* Squeezenet: alexnet-level accuracy with 50x fewer parameters and< 0. 5 MB model size [EB/OL]. (2016) [2018-07-20]. https://arxiv. org/abs/1602. 0736 0. pdf.

[9] Howard A G, Zhu Menglong, Chen Bo, *et al.* Mobilenets: efficient convolutional neural networks for mobile vision applications [EB/OL]. (2017) [2018-07-20]. https://arxiv. org/abs/1704. 04861. pdf.

[10] Sandler M, Howard A, Zhu Menglong, *et al.* Inverted residuals and linear bottlenecks: mobile networks for classification, detection and segmentation [EB/OL]. (2018) [2018-07-20] https://arxiv. org/abs/1801. 04381 v2. pdf.

[11] Hu Jie, Shen Li, Sun Gang. Squeeze-and-excitation networks [EB/OL]. (2017) [2018-07-20]. https://arxiv. org/abs/170 9. 015 07. pdf.

[12] Zhang Xiangyu, Zhou Xinyu, Lin Mengxiao, *et al.* ShuffleNet: an extremely efficient convolutional neural network for mobile devices [EB/OL] (2017) [2018-07-20]. https://arxiv. org/abs/1707. 01 083. pdf.

[13] Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift [EB/OL]. (2015) [2018-07-20]. https://arxiv. org/ab s/1502. 03167. pdf.

[14] Szegedy C, Ioffe S, Vanhoucke V, *et al.* Inception-v4, inception-resnet and the impact of residual connections on learning [C]// Proc of AAAI. Palo Alto, CA: AAAI Press, 2017: 12.

[15] Szegedy C, Vanhoucke V, Ioffe S, *et al.* Rethinking the inception architecture for computer vision [C]// Proc of IEEE conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2016: 2818-2826.

[16] Chollet F. Xception: deep learning with depthwise separable convolutions [EB/OL]. (2016) [2018-07-20]. https://arxiv. org/abs/1610. 02357. pdf.

[17] Xie Saining, Girshick R, Dollár P, *et al.* Aggregated residual transformations for deep neural networks [C]// Proc of Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2017: 5987-59 5995